UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/991,017	11/16/2001	Paul M. Petersen	042390.P12628	2968

7590 11/14/2006

Crystal D. Sayles
c/o BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP
12400 Wilshire Boulevard
Seventh Floor
Los Angeles, CA 90025

EXAMINER	
TRUONG, CAMQUY	

ART UNIT	PAPER NUMBER
2195	

DATE MAILED: 11/14/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/991,017	PETERSEN, PAUL M.	
	Examiner	Art Unit	
	Camquy Truong	2195	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *26 February 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,4-6,8-17 and 19-30* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,4-6,8-17 and 19-30* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1, and 4-6, 8-17, 19-30 are presented for examination. Claims 2-3, 7 and 18 are cancelled.

2. It is noted that although the present application does contain line numbers in the specification and claims, the line numbers in the claims do not correspond to the preferred format. The preferred format is to number each line of every claim, with each claim beginning with line 1. For ease of reference by both the examiner and Applicant all future correspondence should include the recommended line numbering.

Claim Rejections - 35 USC § 112

The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claims 1, 4-6, 8-13, and 21-26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

A. The claim language in the following claims is not clearly understood:

i. As to claims 1 (lines 9-10) and 21(lines 11-12), it is not clearly understood whether " the executed task" refers to the task in the first stack or in the second stack.

Claim Rejections - 35 USC § 103

4. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

5. Claims 1, and 4-6, 8-17, 19-30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Flood et al (U.S. Patent 6, 823, 351) in view of Shavit et al (Patent Application Publication 2003/0005025 A1).

6. Flood and Shavit were cited in the last office action.

7. As to claims 1 and 21, Flood teaches the invention substantially as claimed including a method comprising:

Creating a first stack of tasks associated with a first thread (col. 4, lines 26-28; Fig. 4a, 4b and 4c; col. 6, lines 17-20);

Creating another stack of tasks associated with another thread (col. 4, lines 26-28; Fig. 4a, 4b and 4c; col. 6, lines 17-20);

Executing tasks on the first stack of tasks with the first thread (col. 4, lines 32-35; col. 6, lines 29-30; col.7, lines 31-33);

Determining if the second stack of tasks contains a queued task executable by the first thread by examining a bit mask (col. 4, lines 51-52; col. 7, lines 51-53; col. 9,

lines 2-10; col. 17, line 65 – col. 18, line 14), wherein the bit mask is locked before the bit mask is examined (col. 8, lines 40-46; col. 9, lines 11-13); and

Executing a queued task in the second stack by the first thread (col. 6, lines 31-32; col. 9, lines 2-5).

8. Flood does not explicitly teach that if the executed task is a taskq task, storing any additional tasks generated by the taskq task in the second stack. However, Shavit teaches if the executed task is a taskq task, storing any additional tasks generated by the taskq task in the second stack (paragraph 87- paragraph 88; paragraph 96, lines 11-19; claim 1, lines 11-19).

9. It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the teaching of Flood and Shavit because Shavit's if the executed task is a taskq task, storing any additional tasks generated by the taskq task in the second stack would improve the efficiency of Flood's system by storing any additional tasks generated by the taskq task in the second stack, if the executed task is a taskq task to not only reducing collection-cycle length but also increasing overall efficiency is to segregate the heap into one or more parts, called generations, that are subject to different collection policies.

10. As to claim 14, it is rejected for the same reason as claim 1. In addition, Flood teaches:

Creating a plurality of threads each having a stack of queued tasks (col. 4, lines 26-28; Fig. 4a, 4b and 4c; col. 6, lines 17-20);

At least one thread executing tasks on its stack of queued tasks until no queued task remains in its stack of queued tasks that is executable by the at least one thread, wherein the at least one thread becomes an idle thread (col. 4, lines 32-39; col.6, lines 30-31; col. 7, lines 46-51);

The idle thread searching a bit mask for a bit that is set indicating a thread that may have a task executable by the idle thread (col. 7, lines 51-56; col. 8, lines 16-30; col. 17, line 65- col. 18, lines 10), wherein the bit mask is locked before the bit mask is searched (col. 4, lines 18-20; col. 11, lines 41-51; col.12, lines 11-13);

In response to a set bit in the bit mask, the idle thread searching the stack of queued tasks owned by another thread for an available queued task that can be executed by the idle thread (col. 6, lines 29-32; col. 7, lines 50-55; col. 9, lines 2-10); and

If an available executable task is found, then the idle thread executing the available task (col. 6, lines 29-32; col. 8, line 65- col. 9, line 4).

11. As to claim 27, it reject for the same reason as claims 1 and 21. In addition, Flood teaches:

A memory including a shared memory location (share common memory, col. 5, lines 27-28);

At least one processor to execute at least a first and second parallel thread (col. 4, lines 23-25; col. 5, lines 26-30).

12. As to claim 4, Flood teaches wherein determining if the second stack contains a task executable by the first thread further comprises searching the second stack of tasks to determine if the second stack of tasks contains the task executable by the first thread (col. 8, lines 16-31; col. 17, line 65- col. 18, line 14).

13. As to claim 5, Flood teaches locking the second stack of tasks by the first thread before it is searched (col.11, lines 42-43; col. 12, lines 11-13).

14. As to claims 6 and 15, Flood teaches changing a bit in the bit mask associated with the second thread if a task executable by the first thread is not on the second stack of tasks (col. 9, lines 5-10; col. 18, lines 8-14).

15. As to claims 8, 16 and 24, Shavit teaches changing a bit in a bit mask in response to executing a taskq task, which generates additional tasks (paragraph 61).

16. As to claim 9, Shavit teaches providing a signal to another thread that an additional task was generated (claim 10, lines 1-23).

17. As to claim 10, Food teaches changing the bit in the bit mask includes changing

a bit associated with the second thread indicating the second stack of tasks contains a task executable by the first thread (col. 17, line 65- col. 18, lines 10).

18. As to claim 11, Food teaches executing all executable tasks on the first stack of tasks before determining if the second stack of tasks contains a task executable by the first thread (col. 6, lines 29-34; col. 7, lines 46-53).

19. As to claims 12-13, 20, 26 and 30, Food teaches causing the first thread to enter a wait state if the second stack of tasks does not contain a queued task executable by the first thread (col. 10, lines 58-63; col. 18, lines 3-6).

20. As to claims 17 and 22, Food teaches enabling an idle thread to search its stack of queued tasks for an available task that is executable in response to the setting of a bit in the bit mask (col. 10, lines 30-42; col. 17, line 65- col. 18, lines 10).

21. As to claim 19, Food teaches in response to the idle thread executing the available executable task, the idle thread searching its stack of queued tasks for another available task that is executable (col. 10, lines 38-42).

22. As to claims 23 and 25, Food teaches the bit mask has a bit associated with the second thread and the bit is changed if a queued task is not on the second stack of

tasks (col. 17, line 65- col. 18, lines 10).

23. As to claim 28, Flood determining if a task executable by the first thread is available, the first thread to examines the bit mask to determine if the second stack of tasks contains the available task and then to searches the second stack of tasks for the available task (col. 8, lines 16-31; col. 17, line 65- col. 18, lines 10).

24. As to claim 29, Food teaches the first thread changes a bit in the bit mask associated with the second thread if the first thread executes an available task in the second stack that generates a task (col.18, lines 8-11).

Response to the argument

25. Applicant arguments filed on 8/22/06 had been considered but they are not persuasive. In the remarks applicant argued (1) Flood does not teach or suggest determining if the second stack of tasks contains a task that can be executed by the first thread by examining a bit mask, (2) Flood does not teach or suggest wherein the bit mask is locked before the bit mask is examined. (2) Flood and Shavit do not teach or suggest wherein if the executed task is a taskq task, storing any additional tasks generated by the taskq task in the second stack.

26. Examiner respectfully traverses Applicant's remarks:

As to point (1), Flood teaches to steal from the top of another thread's work queue; the stealing thread first reads that queue's top index (group of bits) to find where its top entry is. It then reads the identified queue entry and reads the bottom index to make sure that bottom index is not less than or the same as the top index, indicate, the stealing thread will not pop the top the queue if the queue is empty (col. 8, lines 16-31; col. 17, line 65 – col. 18, line 13). Therefore, it has been obvious that Flood teaches the stealing thread examine the group of bits to determine if the queue has task to execute. Then, Flood teaches determining if the second stack of tasks contains a task that can be executed by the first thread by examining a bit mask.

As to point (2), Flood teaches the tag field and the top index (a group of bits) that can be accessed in an atomic compare and swap operation (col. 8, lines 19-23). By using atomic instructions, the instructions that represent operation that always run to completion without interruption by another instruction (col. 4, lines 46-49). Therefore, it has been obvious that Flood teaches locking the top index (the group of bits) while read the index. Then, Flood teaches the bit mask is locked before the bit mask is examined.

As to point (3) Shavit teaches identifying tasks that nodes (tasks) A and B, wherein task B contain two namely to the objects that that nodes C and D represent (paragraph 87, lines 1-9 and paragraph 88, lines 1-8). The task C and D are stored in the LIFO stack (Fig. 12, and Fig. 15). Therefore, it has been obvious that Shavit teaches the executed

task is a taskq task, storing any additional tasks generated by the taskq task in the second stack.

27. **THIS ACTION IS MADE FINAL**. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Conclusion

28. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Camquy Truong whose telephone number is (571) 272-3773. The examiner can normally be reached on 8AM – 5PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Meng-Ai An can be reached on 571-272-3756. The fax phone number for the organization where this application or proceeding is assigned is 571-273-3756.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR of Public PAIP. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIP system, contact the Electronic Business Center (EBC) at 866-217-9197(toll-free).

Camquy Truong

November 6, 2006

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 21